Using Radar Odometry on Small Unmanned Aircraft

Eric Quist, *Brigham Young University*

***Abstract*—This paper develops an algorithm to estimate motion using a radar and ground targets. It involves estimating motion using an Extended Kalman Filter (EKF) with an Inertial Measurement Unit (IMU) and a side-looking Synthetic Aperture Radar (SAR) carried on a fixed wing aircraft flying over unknown, flat terrain. The accuracy of the motion estimation is compared to dead reckoning using only the IMU, with truth data being provided by a standard IMU/GPS Kalman filter. Initial results show that over 4.5km of simulated flight, position drift of around 300m resulted, as compared to 2.5km using only the IMU.**

I. Introduction

True Autonomous Navigation by an Unmanned Air Vehicle (UAV) is predicated on the UAVs ability to recognize its position relative to the surrounding environment. Current navigation systems typically use an Inertial Measurement Unit (IMU) in conjunction with a GPS sensor. In these systems, a Kalman Filter (or Extended Kalman Filter) uses the IMU to propagate the vehicles position and GPS to correct the drift introduced by the IMU. Such systems provide accurate position measurement, but are reliant on the reception of GPS positioning. As GPS signals are easily jammed, solutions that don't rely on GPS are necessary for true autonomous navigation.

Several approaches exist for GPS-denied navigation. One option involves using interoceptive sensors such as an IMU to estimate the relative motion of the aircraft. IMUs can be quite accurate, and are often used successfully on inter-ballistic missiles, but that accuracy requires a very heavy and expensive sensor that is often inappropriate for small, unmanned systems. Smaller, cheaper IMUs, such as those typically found on small UAVs, have a large amount of drift, thus limiting the needed accuracy to small windows of time.

To reduce their dependence on IMUs, many algorithms, such as Nister's Visual Odometry (VO) algorithm[9], estimate relative motion using the change in perspective of consecutive camera images. While VO algorithms have been shown to be accurate and can operate real-time[3], [11], they are severely constrained by the range limitations of optical sensors as well as by their dependence on good weather and daytime navigation (or the use of lighting).

Radar, as compared to vision, has better range resolution and is not limited by environmental factors such as time-of-day, fog, or rain. The use of radar for motion estimation from a moving platform has been investigated[14], but the Size, Weight, and Power (SWaP) requirements of these systems, as well as their cost, has limited the scope and availability of such systems.

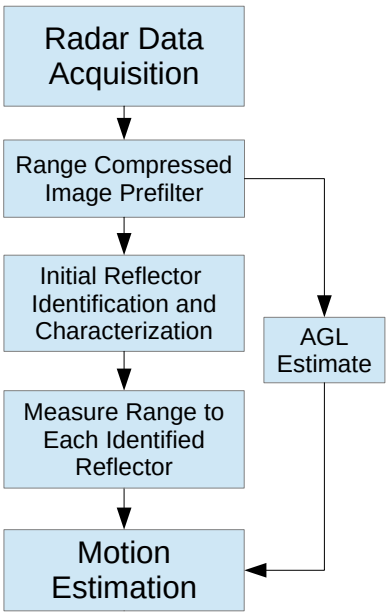


Figure 1. State Estimation Block Diagram

Over the last decade, recent technological advancements have resulted in significant decreases in SWaP for many radar systems, such as Synthetic Aperture Radar (SAR)[12], [4], [13]. Miniaturization allows radar to be an optional payload on small UAVs, and thus considered as an optional sensor to provide motion estimation. Using radar, as opposed to vision, will provide precise range measurements to ground scatterers, while also allowing for night-time and all-weather operation. While some related results have been reported[5], [8], [7], [6], there is currently no working Radar Odometry system.

This paper describes the Radar Odometry system first presented in [10]. It uses radar to limit drift rate of an IMU-based navigation systems.

II. Radar Odometry

The Radar Odometry algorithm is outlined in Figure 1. The algorithm performs an initial range compressed image pre-filter. The filtered image is then used to identify and characterize scatterers with large radar cross section. The range to each identified scatterer is then measured. The height above ground level (AGL) is also estimated using the range compressed image. The aircraft's motion is measured using the range measurements in conjunction with the AGL estimate.

A. Range Compressed Images & Pre-Filtering

There are many approaches to transmitting and receiving radar data. In this paper, we focus on using a Linear Frequency Modulated Continuous Wave (LFM-CW) Radar, which returns the range to scatterers observed in the beamwidth of the radar's single aperture (one antenna).

1) Radar Range and LFM-CW Radar: LFM-CW radar involves repeatedly performing radar transmits and receives, or chirps. The frequency of chirp repetition is referred to as the pulse repetition frequency (PRF),

$$PRF = \frac{1}{\tau},$$

where τ indicates a chirp duration. During a chirp, the LFM-CW radar transmits a single, linear frequency modulated chirp, that starts at time t, and si given by

$$x_t(t) = a_t(t)\cos\left(j2\pi F(t)\,t\right),$$

where

$$a_t(t) = u(t) - u(t-\tau)$$

represents the transmit pulse, $u(t)$ is the unit step function, and where

$$F(t) = F_0 + \frac{\beta}{\tau}t$$

indicates the frequency as a function the initial transmit frequency F_0, transmit bandwidth β, and time t.

After a scatterer reflects the transmitted signal, the reflected signal received by the radar is

$$x_r(t) = a_r(t)\,\sigma_{reflector}x_t(t-\Delta t),$$

where

$$a_r(t) = u(t) - u(t-\tau)$$

is the receive window, $\sigma_{reflector}$ is the scatterer's radar cross section, and Δt is the delay in time between transmit and receive.

The transmit and receive signals are mixed, resulting in

$$\begin{aligned} x_r(t)\otimes x_t(t) &= a_t(t)\cos\left(j2\pi F(t)\,t\right)\cdot \\ &\quad a_r(t)\,\sigma_{reflector}x_t(t-\Delta t) \\ &= a_t(t)\,a_t(t-\Delta)\,a_r(t)\,\sigma_{reflector}\cdot \\ &\quad \cos\left(j2\pi F(t)\,t\right)\cdot \\ &\quad \cos\left(j2\pi F(t-\Delta t)\,(t-\Delta t)\right). \end{aligned}$$

Range compressing a chirp involves performing the Fourier Transform on the mixed transmit and receive signals:

$$X_m(jw) = \mathcal{F}\left(x_r(t)\otimes x_t(t)\right).$$

As the scatterer radar cross-section, and signal strength loss are unknown, and range dependent, the mixed transmit and receive is approximated as

$$\begin{aligned} x_r(t)\otimes x_t(t) &\approx \cos\left(j2\pi F(t)\,t\right)* \\ &\quad \cos\left(j2\pi F(t-\Delta t)\,(t-\Delta t)\right), \end{aligned}$$

resulting in the approximate range compressed signal

$$\begin{aligned} X_m(jw) &\approx \left(\frac{\sin\left(\pi\left(\omega-\frac{\beta}{\tau}\Delta t\right)\tau\right)}{\pi\left(\omega-\frac{\beta}{\tau}\Delta t\right)\tau}\right)^2 \\ &= \operatorname{sync}\left(\omega-\frac{\beta}{\tau}\Delta t\right), \end{aligned}$$

which is the sync function centered at the range-dependent frequency

$$\omega = \frac{\beta}{2\tau}\Delta t = \frac{\beta}{2\tau}\frac{r_i}{c},$$

where r_i is the range to scatterer i and c is the speed of light.

The range compressed chirp represents the accumulative strength of all radar return for a given range r during the specified chip. The mapping from range bin index, b, to range is defined as

$$r = r_0 + br_{res}, \tag{1}$$

where r_0 is the minimum range bin visible to the radar, and r_{res} is the radar's range resolution.

Therefore the radar measurement over the time window $[t-\tau, t]$ of the chirp can be thought of as a column vector where each row index (range bin) represents a particular range, and the value at that index represents the strength of radar return. Since chirps occur sequentially in time, we define the chirp index

$$s = \frac{t-t_0}{\tau}$$

to be the chirp that occurs over the time window $[t_0+(s-1)\,\tau,\ t_0+s\tau]$, where t_0 corresponds to the absolute time the first chirp in the image was started.

Stacking the range measurements at consecutive chirps results ina a positive matrix I_{RC} called the range compressed image as it can be displayed and visualized as an image. The pixel $I_{RC}[b,s]$ is the strength of the radar return from chirp s at range bin b. Figure 2 shows a simulated (i.e., ideal) range compressed image as the radar moves past a single scatterer while traveling a straight line at constant velocity. Due to the antenna's beamwidth and radar parameters, the scatterer is unobserved during some chirps. A more thorough treatment of the range compression derivation may be found in [2].

2) Range-Compressed Image Pre-Filtering: A range compressed image, generated by a LFM-CW Synthetic Aperture Radar, contains a significant amount of noise speckle, as may be demonstrated in Figure 3a. While the scatterers, as seen by the hyperbolic lines, indicate the relative motion of the aircraft, the noise in the imagery limits the ability to distinguish individual scatterers.

To remove the noise, several techniques are implemented. Rather than removing the average pixel value from the entire image, a weighted average pixel value for a each pixel's 9x13 neighborhood is removed from the image. Additionally, as the ranges to each scatterer changes very little in comparison to the chirp index, a weighted horizontal corner kernel,

$$k_{hc} = \begin{bmatrix} -1 & 2 & -1 \end{bmatrix},$$

is also removed from the range compressed image. The weighted image is then thresholded, resulting in the filtered image I_F seen in Figure 3b.

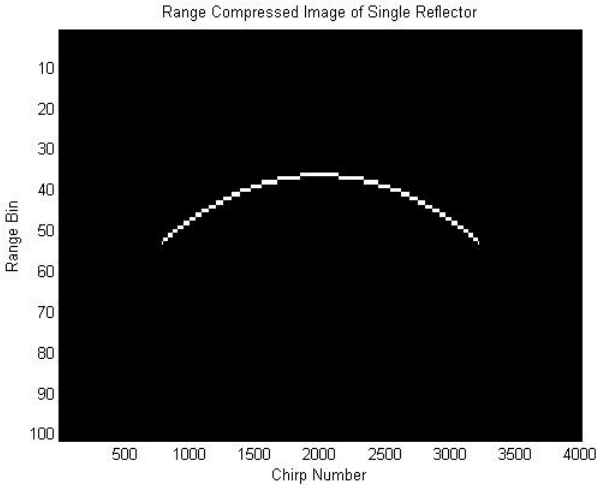
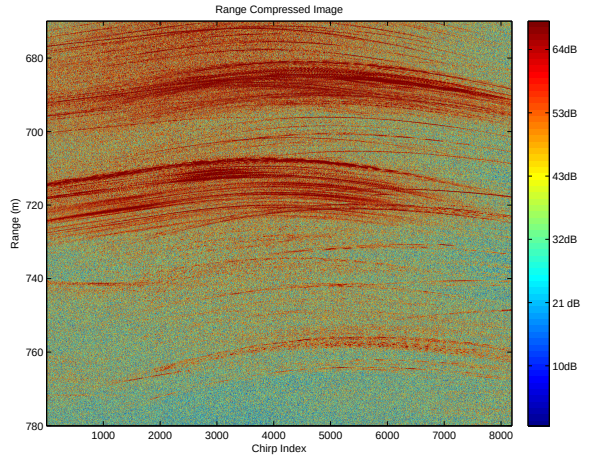
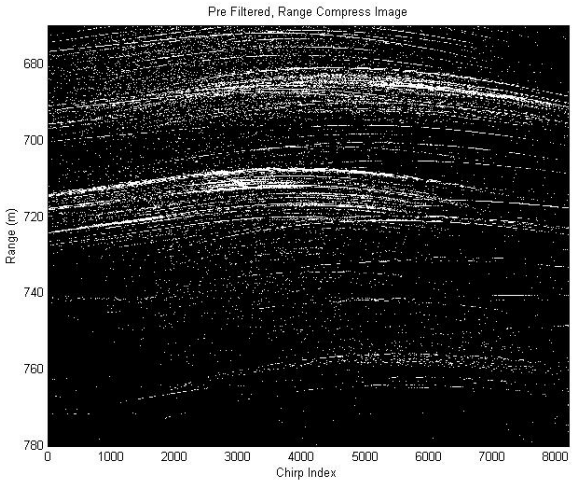


Figure 2. Range compressed image of reflector, with the chirp number corresponding to specific transmit/received pair and the range bin corresponding to the range of the observed scatterer



a)



b)

Figure 3. Range compressed image of radar return during an aircraft flight a)prior to filtering and b)filtered

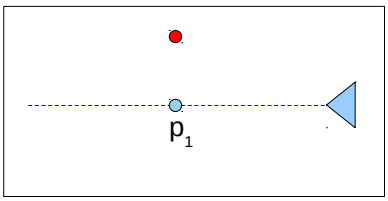


Figure 4. Possible Aircraft Flight Track

B. Initial scatterer Identification

While pre-filtering the range compressed image removes much of the noise, it is still necessary to identify individual scatterers and estimate the aircraft's range to each scatterer. Initially, this is performed by identifying individual scatterers and performing a first-order estimate of the motion of the aircraft relative to each scatterer.

1) Range to scatterer during flight: Consider Figure 4, which shows an aircraft, represented by a triangle, as it flies in a straight line past scatterer i, represented by the red circle, located at $\mathbf{p}_i$. The position of the aircraft $\mathbf{p}_a(t)$, when it is closest to $\mathbf{p}_i$, as identified by the blue circle, is defined as $\mathbf{p}_{i,min}$ and occurs at time $t_{i,min}$. Defining the aircraft position as a function of $t_{i,min}$ results in

$$\mathbf{p}_a(t) = \mathbf{p}_{i,min} + \dot{\mathbf{p}}_a(t)(t - t_{i,min}).$$

The range to scatterer i is represented by $r_i(t)$, while the squared range is calculated as

$$\begin{aligned} r_i(t)^2 &= (\mathbf{p}_a(t) - \mathbf{p}_i)^T (\mathbf{p}_a(t) - \mathbf{p}_i) \\ &= (\mathbf{p}_{i,min} - \mathbf{p}_i)^2 \\ &\quad -2\dot{\mathbf{p}}_a(t)(t - t_{i,min})(\mathbf{p}_{i,min} - \mathbf{p}_i) \\ &\quad +\dot{\mathbf{p}}_a(t)^2 (t - t_{i,min})^2. \end{aligned}$$

The straight flight and fixed velocity assumptions imply that $\dot{\mathbf{p}}_a(t)$ and $\mathbf{p}_{i,min} - \mathbf{p}_i$ are terms becoming orthogonal, resulting in

$$r_i(t)^2 = (\mathbf{p}_{i,min} - \mathbf{p}_t)^2 + \|\dot{\mathbf{p}}_a\|(t - t_{i,min})^2.$$

Further defining the minimum range to scatterer i as $r_{i,min}$ and the aircraft speed $V_g = \|\dot{\mathbf{p}}_a\|$, results in a hyperbolic equation for the range equation:

$$\frac{1}{r_{i,min}^2} r_i^2(t) - \frac{V_g^2}{r_{i,min}^2}(t - t_{i,min})^2 = 1. \tag{2}$$

To express this equation in range compressed image coordinates, let

$$r_i[s] = r_0 + b_i[s] r_{res},$$

which is a function of the the range bin $b_i[s]$ during chirp s. The corresponding minimum range equation is

$$r_{i,min} = r_0 + b_{i,min} r_{res}.$$

Similarly define $t_{i,min}$ and $s_{i,min}$ so that

$$t_{i,min} = s_{i,min}\tau - t_0.$$

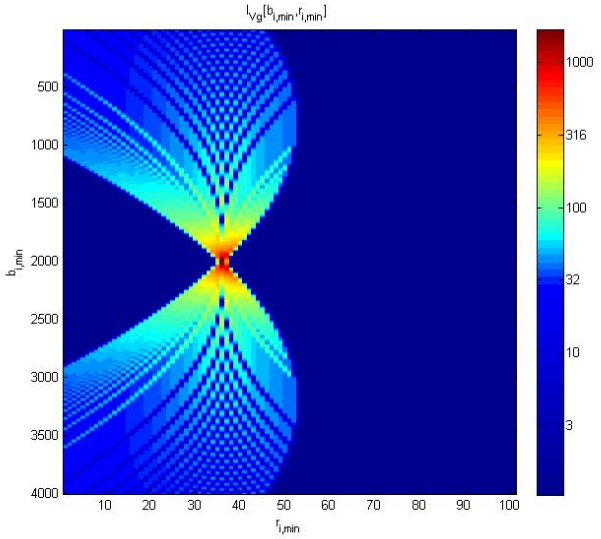


Figure 5. Sample 2-D Hough for a fixed V_g, and color indicates the number of votes

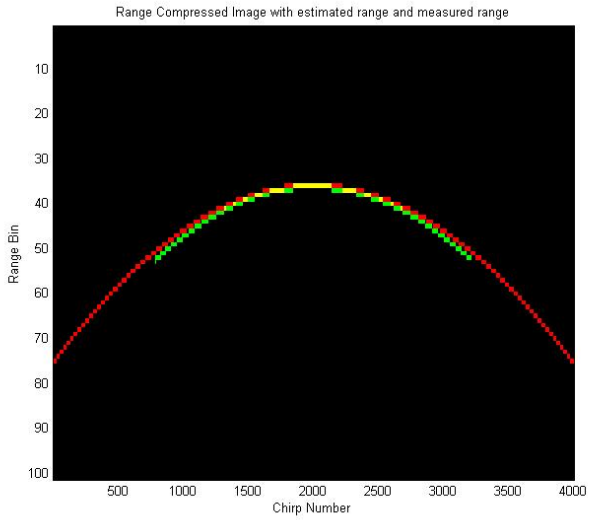


Figure 6. Range compressed image of an observed single scatterer (seen in green) as compared to the initial hyperbola estimate (seen in red)

The hyperbolic range equation using the discrete range bin and chirp number becomes

$$\frac{\left(r_0 + b_i\left[s\right] r_{res}\right)^2}{\left(r_0 + b_{i,min} r_{res}\right)^2} - \frac{V_g^2\left(s - s_{i,min}\right)^2 \tau^2}{\left(r_0 + b_{i,min} r_{res}\right)^2} = 1. \qquad (3)$$

2) Scatterer Detection Using the Hough Transform: As r_0 and r_{res} are constant, and predefined, each scatterer is uniquely defined by V_g, $b_{i,min}$, and $s_{i,min}$ (see Equation 3) which constrain the hyperbolic shape of the scatterer in the range compressed image. A hyperbolic Hough Transform[1] is used to identify, and provide initial parametrization for each scatterer. This is performed by creating a parameter space

$$H\left(V_g,\, b_{i,min},\, s_{i,min}\right)$$

large enough to contain hyperbola parameter combinations observable in the range compressed image. Specifically, the Hough Transform iterates over pixels in the pre-filtered image. When a pixel is illuminated, it votes for all possible parameter combinations that would result in the specific pixel being illuminated. Combinations of parameters that received large numbers of votes suggest that a hyperbola is present with those parameters.

Ideally, when all illuminated pixels have been traversed, the parameter combinations that receive the largest number of votes would be used to parametrize and identify scatterer hyperbolas found in the image. However, non-straight flight, non-constant airspeed, measurement inaccuracies, and the resolution of hyperbola constraints often result in a large number of votes being cast for incorrect constraints. This is illustrated in the Figure 5, which shows the Hough space image for a single, fixed V_g. As can be seen, neighboring range bins and chirp numbers often have similar numbers of votes. To this end, some additional filtering is necessary for initial scatterer estimation.

To further isolate individual hyperbola constraints, a ground velocity $\hat{V}_g$ is estimated to be the ground velocity in the Hough-space cube that contains the pixel with the largest number of votes. The resulting two dimensional Hough-space image $I_{V_g}\left(b_{i,min}, s_{i,min}\right) = H\left(\hat{V}_g, b_{i,min}, s_{i,min}\right)$ is then used to identify specific scatterers. This is done by removing the average pixel value in I_{V_g} is removed from the image. The resulting image I_{mr} is normalized, such that $I_n = \frac{I_{mr}}{\max_{b,s}(I_{mr})}$, then thresholded, resulting in $I_m = I_n > T_n$, where T_n is the threshold applied. The pixels illuminated in the thresholded image are then segmented into connected groups G_{V_g}.

Erroneous peaks are removed from I_{V_g} by convolving the image using a 5x5 smoothing kernel. The pixel with the maximum value from each group $g \in G_{V_g}$ is identified as a scatterer and characterized by Hough-space indexing parameters $\hat{V}_g$, $\hat{b}_{i,min}$, and $\hat{s}_{i,min}$.

C. Range estimation

The initial scatterer identification and parametrization calculated by the Hough Transform provides a rough estimate as to where each scatterer is at each chirp. While imprecise, this estimate provides a starting point to determine a more precise range estimate for each chirp.

Consider Figure 6 which shows a range compressed image with the measured range to the scatterer seen in green, as compared to the initial hyperbola range estimate seen in red. Estimating the range to the scatterer i involves starting at the focus of the hyperbola and estimating the difference, d_i, between the initial estimate, $\hat{b}_i$, and the measured range, b_i, to the scatterer at each chirp. Additionally, an uncertainty estimate e_i is calculated, which identifies the number of chirps, starting at $\hat{s}_{i,min}$, over which the range to the scatterer was not observed.

The uncertainty estimate and drift estimates are initialized to 0 for all chirps. Starting at

$$s_{inc} = \hat{s}_{i,min} + 1$$

Equation 3 is used to estimate the range $\hat{b}_i[s_{inc}]$, for each incremental s_{inc}. If the pixel in the filtered image $I_F\left(\hat{b}_i[s_{inc}] + d_i[s_{inc}-1], s_{inc}\right)$ is illuminated or if $I_F\left(\hat{b}_i[s_{inc}] + d_i[s_{inc}-1] + 1, s_{inc}\right)$ and $I_F\left(\hat{b}_i[s_{inc}] + d_i[s_{inc}-1] - 1, s_{inc}\right)$ are illuminated, the difference is propagated as

$$d_i[s_{inc}] = d_i[s_{inc}-1], \tag{4}$$

the uncertainty is propagated as

$$e_i[s_{inc}] = e_i[s_{inc}-1], \tag{5}$$

and the range is estimated as

$$b_i[s_{inc}] = \hat{b}_i[s_{inc}] + d_i[s_{inc}]. \tag{6}$$

Otherwise, the neighboring chirp pixels are evaluated ($I_F\left(\hat{b}_i[s_{inc}] + d_i[s_{inc}-1] + 1, s_{inc}\right)$ and $I_F\left(\hat{b}_i[s_{inc}] + d_i[s_{inc}-1] - 1, s_{inc}\right)$). If either of them is illuminated, the difference is incremented as

$$d_i[s_{inc}] = d_i[s_{inc}-1] + 1,$$

or decremented, such that $d_i[s_{inc}]$ would result in $I_F\left(\hat{b}_i[s_{inc}] + d_i[s_{inc}], s_{inc}\right)$ being illuminated. The uncertainty is again propagated, and the range is estimated as in Equation 6.

In the event the range remains unassigned, the uncertainty is incremented as

$$e_i[s_{inc}] = e_i[s_{inc}-1] + 1,$$

the difference is propagated as in Equation 4, and the range is estimated using Equation 6.

When all chirps after $\hat{s}_{i,min}$ have been assessed, the chirp index is assigned to

$$s_{inc} = \hat{s}_{i,min} - 1$$

and the same process is repeated, for each decreasing chirp index s_{inc}. When decrementing the chirp index, the difference and uncertainty continue to be propagated using the previously evaluated index $s_{inc}+1$.

The algorithm results in a range estimate to each scatterer, for each chirp, along with an uncertainty metric which indicates the number of unobservable states that were reached to obtain the range estimate. To identify scatterers that are no longer observable, when the uncertainty is higher than a certain threshold, the scatterer is considered unobserved. Figure 7 shows the difference, uncertainty, initial range estimate, and measured range for the range compressed image seen in Figure 6.

D. AGL Estimation

Nadir (the return from the ground immediately below the UAV) is used to estimate the AGL. This is calculated as the first range bin in the range compressed image with a signal larger than the AGL threshold T_{AGL}, and is identified as d_{AGL}. The threshold is present to remove the measurement noise.

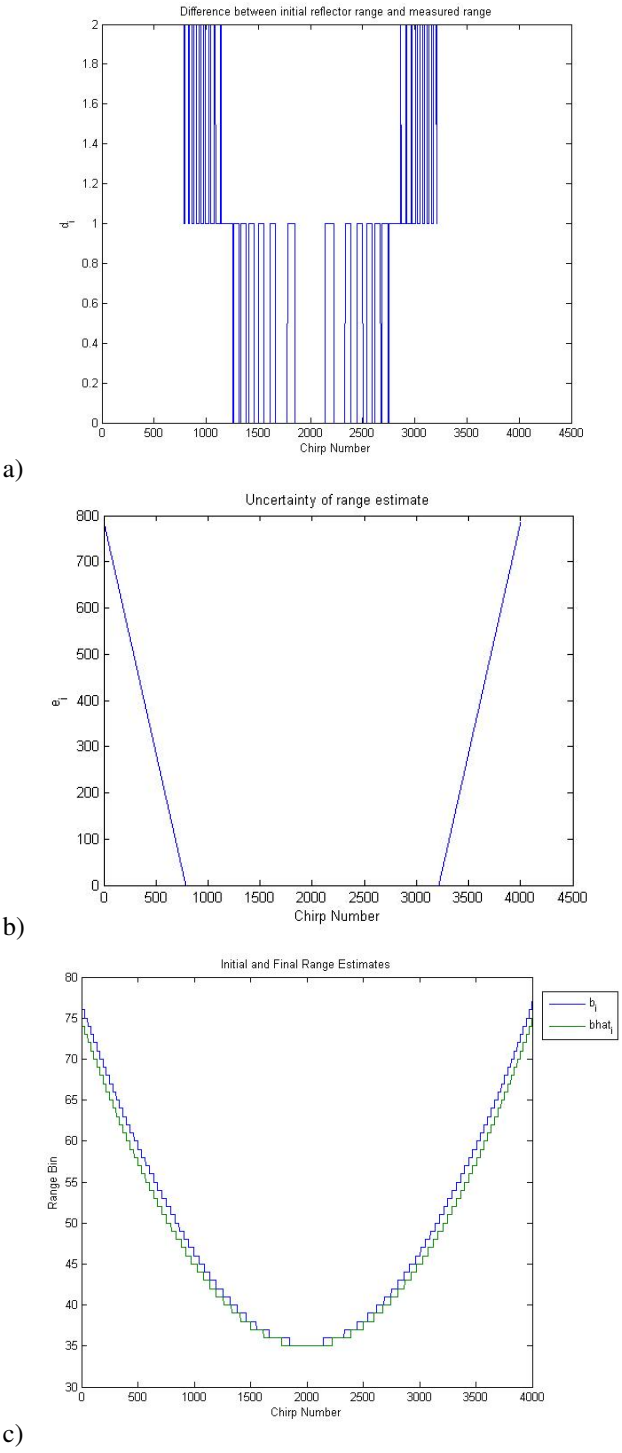


Figure 7. a)Difference and b)uncertainty for observed scatterer, along with the c) initial range estimate (seen in green) and measured range estimate (shown in blue).

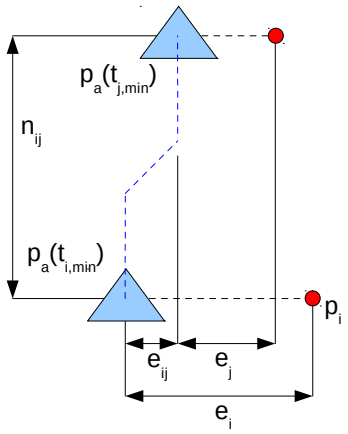


Figure 8. Motion experienced by an aircraft between times $t_{i,min}$ and $t_{j,min}$

E. Motion Estimation

Using the estimated range to multiple scatterers, the motion of the aircraft is estimated. This is performed by calculating the incremental motion using the estimated range to consecutive pairs of scatterers. The ranges to each scatterer, i and j, are both observed at $t_{i,min}$ and $t_{j,min}$. For simplicity, a northern flight-track is assumed. A north-east-down coordinate system is used, with down being relative to ground-level.

Define the position of the aircraft at $t_{i,min}$ as

$$\mathbf{p}_a\left(t_{i,min}\right) = \begin{bmatrix} n_{i,min} \\ e_{i,min} \\ d_{i,min} \end{bmatrix}.$$

At time $t_{j,min}$, the position of the aircraft may be represented as

$$\mathbf{p}_a\left(t_{j,min}\right) = \begin{bmatrix} n_{ij} + n_{i,min} \\ e_{ij} + e_{i,min} \\ d_{j,min} \end{bmatrix},$$

where n_{ij} and e_{ij} indicate the northern and eastern aircraft motion between $t_{i,min}$ and $t_{j,min}$. A flat-earth model is assumed. Figure 8 visualizes the aircraft at times $t_{i,min}$ and $t_{j,min}$. It also shows scatterer i, located at

$$\mathbf{p}_i = \begin{bmatrix} n_{i,min} \\ e_i + e_{i,min} \\ 0 \end{bmatrix},$$

and scatterer j, located at

$$\mathbf{p}_j = \begin{bmatrix} n_{ij} + n_{i,min} \\ e_j + e_{ij} + e_{i,min} \\ 0 \end{bmatrix}.$$

Assuming that both scatterers are visible at times $t_{i,min}$ and $t_{j,min}$, the squared range to the scatterers are

$$\begin{aligned} r_i^2\left(t_{i,min}\right) &= e_i^2 + d_{i,min}^2 = r_{i,min}^2 \\ r_j^2\left(t_{i,min}\right) &= n_{ij}^2 + \left(e_j + e_{ij}\right)^2 + d_{i,min}^2 \\ r_i^2\left(t_{j,min}\right) &= n_{ij}^2 + \left(e_i - e_{ij}\right)^2 + d_{j,min}^2 \\ r_j^2\left(t_{j,min}\right) &= e_j^2 + d_{j,min}^2 = r_{j,min}^2. \end{aligned}$$

The down position of the aircraft is provided by the AGL measurement,

$$\begin{aligned} d_{i,min} &= d_{AGL}\left(t_{i,min}\right) \\ d_{j,min} &= d_{AGL}\left(t_{j,min}\right) \end{aligned}$$

which allows for the eastern position term for each target to be calculated as

$$\begin{aligned} e_i &= \sqrt{r_{i,min}^2 - d_{AGL}\left(t_{i,min}\right)^2} \\ e_j &= \sqrt{r_{j,min}^2 - d_{AGL}\left(t_{j,min}\right)^2}. \end{aligned}$$

Subtracting the cross-terms results in

$$r_j^2\left(t_{i,min}\right) - r_i^2\left(t_{j,min}\right) = e_j^2 - d_{j,min}^2 - e_i^2 + d_{i,min}^2 + 2e_{ij}\left(e_j + e_i\right).$$

Solving for e_{ij} gives

$$e_{ij} = \frac{r_j^2\left(t_{i,min}\right) - r_i^2\left(t_{j,min}\right) - e_j^2 + d_{j,min}^2 + e_i^2 - d_{i,min}^2}{2\left(e_j + e_i\right)}.$$

The northern motion may also be as solved as

$$n_{ij} = \sqrt{r_i\left(t_{j,min}\right)^2 - \left(e_i - e_{ij}\right)^2 - d_{j,min}^2},$$

which allows for the aircraft motion to be calculated using the measured ranges $r_i\left(t_{i,min}\right)$, $r_j\left(t_{i,min}\right)$, $r_i\left(t_{j,min}\right)$, and $r_i\left(t_{j,min}\right)$ in addition to the measured AGL $d_{AGL}\left(t_{i,min}\right)$ and $d_{AGL}\left(t_{j,min}\right)$.

Selecting scatterer pairs p_i and p_j such that the scatterers are both visible at $t_{i,min}$ and $t_{j,min}$ involves sorting the scatterers by their respective t_{min}. Sequential scatterers are then selected and the resulting northern and eastern motion is calculated using each scatterer pair.

III. Kalman Filter

The radar motion estimation algorithm does not take advantage of other available sensors. Using IMUs in conjunction with the radar provides additional accuracy. Combining the multiple sensors with different update rates is often performed by a Kalman Filter, or an Extended Kalman Filter (EKF) when the system has non-linear dynamics.

A. Sensor Models

1) IMU: An IMU sensor consists of both accelerometers and rate gyros. Each of the three accelerometers measure the acceleration along its axis, with each accelerometer typically aligned with one of the body-frame axis, resulting in

$$\begin{aligned} y_{accel,x} &= a_x + \eta_{accel,x} \\ y_{accel,y} &= a_y + \eta_{accel,y} \\ y_{accel,z} &= a_z + \eta_{accel,z}, \end{aligned}$$

where a_x, a_y, and a_z represent the acceleration, and $\eta_{accel,x}$, $\eta_{accel,y}$, and $\eta_{accel,z}$ represent the noise, each along its specified axis.

The three rate gyros, also aligned with the body-frame axes, measure rotation around the specific axis,

$$\begin{aligned} y_{gyro,x} &= p + \eta_{gyro,x} \\ y_{gyro,y} &= q + \eta_{gyro,y} \\ y_{gyro,z} &= r + \eta_{gyro,z}, \end{aligned}$$

where $\eta_{gyro,x}$, $\eta_{gyro,y}$, and $\eta_{gyro,z}$ are the noise along each axis.

2) Radar Odometry: The Radar Odometry approach measures the along-track and cross-track velocity over time. For simplicity, flight is considered in a straight north direction, resulting in

$$\begin{aligned} y_{RO,n} &= \dot{n} + \eta_{RO,n} \\ y_{RO,e} &= \dot{e} + \eta_{RO,e} \\ y_{RO,AGL} &= -d + \eta_{RO,AGL}. \end{aligned}$$

B. Prediction Models

For the system under consideration, position

$$\mathbf{p} = \begin{bmatrix} n \\ e \\ d \end{bmatrix}$$

and velocity

$$\mathbf{v} = \begin{bmatrix} \dot{n} \\ \dot{e} \\ \dot{d} \end{bmatrix}$$

are represented using the inertial frame, while attitude is represented as

$$\mathbf{\Theta} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix},$$

where ϕ is the roll angle, θ is the pitch angle, and ψ is the heading angle.

The system state is given by

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{\Theta} \end{bmatrix}.$$

with the state dynamics

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}).$$

The input for the prediction step is the gyro and accelerometer,

$$\mathbf{u} = \begin{bmatrix} \mathbf{a}^b \\ \omega \end{bmatrix},$$

resulting in the full state transition model

$$f(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \mathbf{v} \\ \mathbf{g} + R_b^i(\Theta)\,\mathbf{a}^b \\ S(\Theta)\,\omega \end{bmatrix},$$

where $R_b^i(\Theta)$ is the body to inertial frame rotation and $S(\Theta)$ is

$$S(\Theta) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix}.$$

The Jacobian of $f(\mathbf{x}, \mathbf{u})$ is given by

$$\frac{\partial f}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 0^{3x3} & I^{3x3} & 0^{3x3} \\ 0^{3x3} & 0^{3x3} & \frac{\partial R_b^i(\Theta)\mathbf{a}^b}{\partial \Theta} \\ 0^{3x3} & 0^{3x3} & \frac{\partial S(\Theta)\omega}{\partial \Theta} \end{bmatrix},$$

where

$$\frac{\partial R_b^i(\Theta)\,\mathbf{a}^b}{\partial \Theta} = a_x \begin{bmatrix} 0 & -\mathrm{s}_\theta\mathrm{c}_\psi & -\mathrm{c}_\theta\mathrm{s}_\psi \\ 0 & -\mathrm{s}_\theta\mathrm{s}_\psi & \mathrm{c}_\theta\mathrm{c}_\psi \\ 0 & -\mathrm{c}_\theta & 0 \end{bmatrix} + a_y \begin{bmatrix} \mathrm{c}_\phi\mathrm{s}_\theta\mathrm{c}_\psi + \mathrm{s}_\phi\mathrm{s}_\psi & \mathrm{s}_\phi\mathrm{c}_\theta\mathrm{c}_\psi & -\mathrm{s}_\phi\mathrm{s}_\theta\mathrm{s}_\psi - \mathrm{c}_\phi\mathrm{c}_\psi \\ \mathrm{c}_\phi\mathrm{s}_\theta\mathrm{s}_\psi - \mathrm{s}_\phi\mathrm{c}_\psi & \mathrm{s}_\phi\mathrm{c}_\theta\mathrm{s}_\psi & \mathrm{s}_\phi\mathrm{s}_\theta\mathrm{c}_\psi - \mathrm{c}_\phi\mathrm{s}_\psi \\ \mathrm{c}_\phi\mathrm{c}_\theta & -\mathrm{s}_\phi\mathrm{s}_\theta & 0 \end{bmatrix} + a_y \begin{bmatrix} -\mathrm{s}_\phi\mathrm{s}_\theta\mathrm{c}_\psi + \mathrm{c}_\phi\mathrm{s}_\psi & \mathrm{c}_\phi\mathrm{c}_\theta\mathrm{c}_\psi & -\mathrm{c}_\phi\mathrm{s}_\theta\mathrm{s}_\psi + \mathrm{s}_\phi\mathrm{c}_\psi \\ -\mathrm{s}_\phi\mathrm{s}_\theta\mathrm{s}_\psi - \mathrm{c}_\phi\mathrm{c}_\psi & \mathrm{c}_\phi\mathrm{c}_\theta\mathrm{s}_\psi & \mathrm{c}_\phi\mathrm{s}_\theta\mathrm{c}_\psi + \mathrm{s}_\phi\mathrm{s}_\psi \\ -\mathrm{s}_\phi\mathrm{c}_\theta & -\mathrm{c}_\phi\mathrm{s}_\theta & 0 \end{bmatrix},$$

and

$$\frac{\partial S(\Theta)\,\omega}{\partial \Theta} = \begin{bmatrix} q\mathrm{c}_\phi\mathrm{t}_\theta - r\mathrm{s}_\phi\mathrm{t}_\theta & q\frac{\mathrm{s}_\phi}{\mathrm{c}_\theta^2} + r\frac{\mathrm{c}_\phi}{\mathrm{c}_\theta^2} & 0 \\ -q\mathrm{s}_\phi - r\mathrm{c}_\phi & 0 & 0 \\ q\frac{\mathrm{c}_\phi}{\mathrm{c}_\theta} - r\frac{\mathrm{s}_\phi}{\mathrm{c}_\theta} & q\frac{\mathrm{s}_\phi\mathrm{s}_\theta}{\mathrm{c}_\theta^2} + r\frac{\mathrm{c}_\phi\mathrm{s}_\theta}{\mathrm{c}_\theta^2} & 0 \end{bmatrix}.$$

C. Measurement Model

1) GPS Measurement Model: When using GPS, the update model is

$$h_{GPS}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} n \\ e \\ -d \\ V_g \\ \psi \end{bmatrix},$$

with the associated Jacobian

$$\frac{\partial h_{GPS}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\dot{n}}{\sqrt{\dot{n}^2+\dot{e}^2}} & \frac{\dot{e}}{\sqrt{\dot{n}^2+\dot{e}^2}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{-\dot{e}}{\dot{e}^2+\dot{n}^2} & \frac{\dot{n}}{\dot{e}^2+\dot{n}^2} & 0 & 0 & 0 & 0 \end{bmatrix}.$$

2) Radar Odometry Update Model: Radar Odometry provides two measurement update models. The AGL update is acquired every 100 milliseconds and may be measured regardless of the measured scatterers. It's update model is of the form

$$h_{RO,AGL}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} -d \end{bmatrix},$$

with the Jacobian

$$\frac{\partial h_{RO,AGL}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

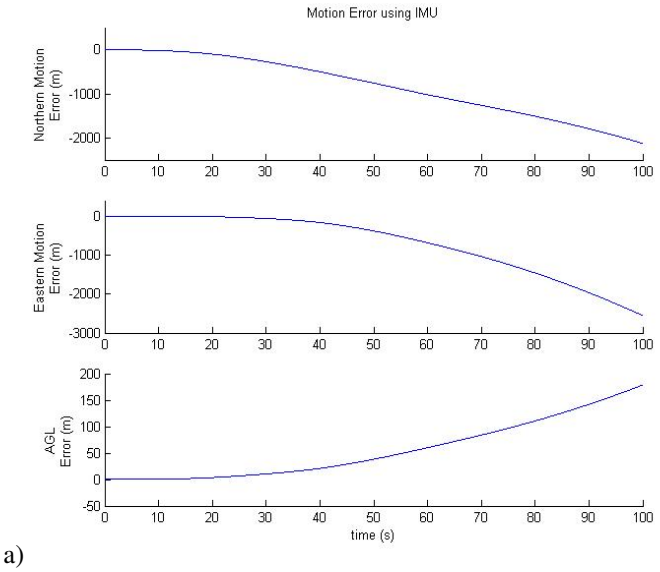
To measure ground velocity, the Radar Odometry update model is

$$h_{RO,V_{ground}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{n} \\ \dot{e} \end{bmatrix},$$

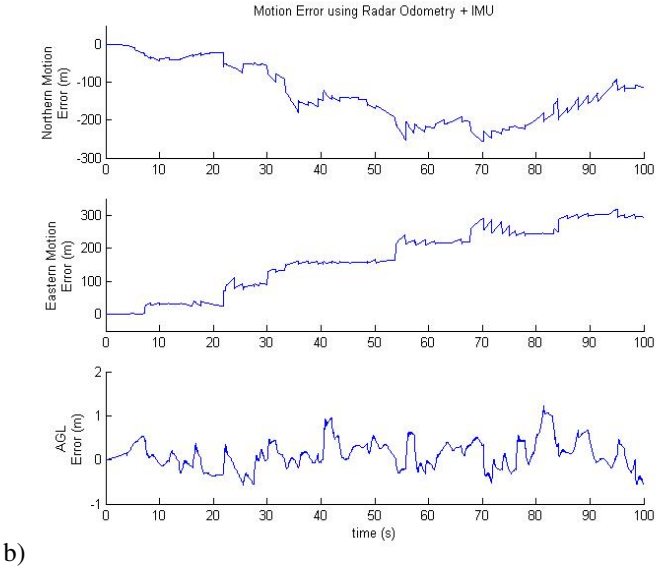
with the associated Jacobian

$$\frac{\partial h_{RO,V_{ground}}}{\partial \mathbf{x}}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The ground velocity measurement is updated as often as the algorithm is able to correctly measure ground motion.



a)



b)

Figure 9. Position error over time using a)IMU and b)IMU and RO

IV. Results

We simulated an unmanned aircraft over an unknown, flat terrain using an IMU alone and using both IMU and Radar Odometry sensors. The simulations used the IMU to propagate the aircraft's pose, while the Radar Odometry algorithm, when used, provided a corrective update term. In all tests, simulated flight dynamics, wind, and sensor noise were implemented. Each test started with an aircraft flying at a fixed, known velocity at a known location.

Figure 9 shows the position error from a 100 second simulation, with the aircraft flying 45 m/s. As expected, the IMU-only solution (Figure 9a) has a large drift rate, resulting in northern position error of up to 2500m, with an eastern position error reaching around 3000m, and AGL error of 200m. Radar Odometry and IMU (Figure 9b) resulted in a significantly smaller drift rate, with worse-case along-track error of 300m, worse-case cross-track error of 300m, and AGL error of less than 2 meters.

V. Conclusion

When the GPS signal is lost or denied, current small UAV systems are unable to accurately estimate their position. Using the Radar Odometry algorithm derived in this paper in simulation has shown to reduce drift to less than 7%, as compared to the 66% drift from an IMU alone. Such results suggest that the using radar for motion estimation is a feasible alternative to traditional vision systems, particularly when using fixed-wing aircraft outdoors. Future work involves handling non-straight flight-tracks and minimizing the error in the current approach. We hope to integrate this solution with other placement recognition work to create a complete solution to GPS-denied navigation.

References

[1] Richard O Duda and Peter E Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972.

[2] Michael I Duersch. *BYU Micro-SAR: A Very Small, Low-Power LFM-CW Synthetic Aperture Radar*. Master's, Brigham Young University, 2004.

[3] F Fraundorfer and D Scaramuzza. Visual Odometry: Part II: Matching, Robustness, Optimization, and Applications. *IEEE Robotics & Automation Magazine*, 19(2):78–90, 2012.

[4] Dmitriy Garmatyuk, Jonathan Schuerger, Kyle Kauffman, and Scott Spalding. Wideband OFDM system for radar and communications. *2009 IEEE Radar Conference*, pages 1–6, 2009.

[5] Ebi Jose, Martin Adams, Senior Member, John Stephen Mullane, and Nicholas M. Patrikalakis. Predicting Millimeter Wave Radar Spectra for Autonomous Navigation. *IEEE Sensors Journal*, 10(5):960–971, 2010.

[6] Kyle Kauffman, Yu Morton, John Raquet, and Dmitriy Garmatyuk. Simulation Study of UWB-OFDM SAR for Dead-Reckoning Navigation. In *Institute of Navigation*, pages 153–160, 2010.

[7] Kyle Kauffman and John Raquet. Enhanced feature detection and tracking algorithm for UWB-OFDM SAR navigation. *Proceedings of the 2011 IEEE National Aerospace and Electronics Conference (NAECON)*, pages 261–269, July 2011.

[8] Dirk Langer. An integrated MMW radar system for outdoor navigation. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, number April, pages 417–422, 1996.

[9] David Nistér, Oleg Naroditsky, and James Bergen. Visual Odometry. In *Computer Vision and Pattern Recognition 2004*, volume 11, January 2004.

[10] Eric B. Quist and Randal W. Beard. Radar Odometry on Small Unmanned Aircraft. In *Submitted GNC 2013*, pages 1–18, 2013.

[11] Davide Scaramuzza and Friedrich Fraundorfer. Visual Odometry: Part I: The First 30 Years and Fundamentals. *IEEE Robotics & Automation Magazine2*, 18(December):80–92, 2011.

[12] David Schneider. Winner : Radio Eye in the Sky. *IEEE Spectrum*, 2009.

[13] Bruce Walker and Grant Sander. A high-resolution, four-band SAR Testbed with real-time image formation. In *Geoscience and Remote Sensing Symposium, 1996*, volume 0529, pages 1881–1885, 1996.

[14] Richard N. White. Airborne Doppler Radar Navigation of Jet Transport Aircraft. *Aerospace and Navigational Electronics, IRE Transactions on*, pages 11–20, 1962.